UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/017,793	12/12/2001	Youfeng Wu	042390.P12589	9134

7590 01/13/2005

Crystal D. Sayles
c/o BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP
12400 Wilshire Boulevard, Seventh Floor
Los Angeles, CA 90025

EXAMINER	
GERSTL, SHANE F	

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 01/13/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application N .	Applicant(s)	
	10/017,793	WU ET AL.	
	Examin r	Art Unit	
	Shane F Gerstl	2183	

-- The MAILING DATE of this communicati n appears on the cover sheet with the c rresp ndence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *28 October 2004 and 22 December 2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-27* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-27 have been examined.

Papers Received

2. Receipt is acknowledged of change in power of attorney paper submitted, where the paper has been placed of record in the file.

3. The amendment has successfully overcome the 35 USC 101 rejections, which are herein withdrawn.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

5. Claims 1, 4, 6, 7, 9-13, 15, 17-20, 22-24, 26, and 27 are rejected under 35 U.S.C. 102(e) as being anticipated by Mukherjee (6,757,811).

6. In regard to claim 1, Mukhejree discloses a data processing apparatus, comprising:

a. a first pipeline (fig. 2) having a data cache (element 146) and an instruction cache (element 110);

b. a second pipeline coupled to the data cache and the instruction cache;

[Column 3, line 66 – column 4, line 10 and column 5, lines 26-29 show that the

processor is a simultaneously redundant processor ("SRT", a type of simultaneous multithreaded processor that executes the same code in each thread) and thus there are at least two pipelines that execute the instructions in parallel as illustrated by figure 2 where the execution units 134, 138, and 142 are each replicated for such operation.]

c. and data value prediction module coupled to the second pipeline. [Column 6, line 59 – column 7, line 29 show that the SRT pipelines are separated by "slack" and thus the redundant threads comprise a leading (2^{nd}) and trailing (1^{st}) thread or pipeline. Column 7, lines 52-59 show that upon a [data] cache miss in the leading thread or pipeline, the cache is updated so it will not miss in the trailing thread. This means that when a load instruction misses the cache in the leading thread the load value is retrieved from memory and updates the data cache. Thus a the trailing thread predicts that the trailing thread will use the same load value and updates the cache accordingly using an inherent module, which is appropriately called a data value prediction module. The given definition shows that to predict is to "make known in advance". Thus the leading thread by making the data available in the cache ahead of time or make the data know in advance by placing it in the cache for the trailing thread and thus the second pipeline makes this data known in advance or predicts this data.]

7. In regard to claim 4, Mukherjee discloses the data processing apparatus of claim 1, further comprising: a first register file coupled to the first pipeline; and a second register file coupled to the second pipeline. [Figure 2 shows that an FP register file and

an integer register file are each coupled to both pipelines and thus a first register file is coupled to the first pipeline and a second register file to the 2nd pipeline.]

8. In regard to claim 6, Mukherjee discloses the data processing apparatus of claim 1, wherein the first and second pipelines are included in a single processor (figure 2).

9. In regard to claim 7, Mukherjee discloses the data processing apparatus of claim 6, wherein the data cache, the instruction cache, and the data value prediction module are included in the single processor (figure 2).

10. In regard to claim 9, Mukherjee discloses the data processing apparatus of claim 1, further comprising: a main memory (figure 1, element 92) coupled to the data cache, wherein the first pipeline may operate to store a data value to the main memory, and wherein the second pipeline may not operate to store the data value to the main memory. [As shown above, the pipelines are staggered and thus if one pipeline is executing a store (as explained in the sections cited above), the other pipeline may not be executing a store.]

11. In regard to claim 10, Mukherjee discloses the data processing apparatus of claim 1, further comprising: a storage buffer coupled to the second pipeline. [Figure 2, illustrates a storage buffer as element 130. Column 6, lines 15-23 show that this unit acts as an instruction queue or buffer.]

12. In regard to claim 11, Mukherjee discloses the data processing apparatus of claim 1, further comprising: a synchronization mechanism coupled to the second pipeline. [Column 7, lines 7-29 show that a slack counter is used to keep a delay between the two pipelines (threads) or synchronize them by a certain delay.]

13. In regard to claim 12, Mukherjee discloses the data processing apparatus of claim 11, wherein the synchronization mechanism includes a misprediction counter. [As shown above, the synchronization mechanism includes a slack counter. Column 7, lines 30-51 show that the slack provided by the counter allows for branch misspeculation or misprediction to be avoided in the trailing thread and thus an adequate name for this counter would also be a mispredict counter since the counter gives sufficient time to handle a misprediction.]

14. In regard to claim 13, Mukherjee discloses a computer, comprising:

a. a first processor (figure 1, element 100) including a first pipeline having a data cache (figure 2, element 146) coupled to a memory (figure 1, element 92), and an instruction cache (figure 2, element 110);

b. a second pipeline coupled to the data cache and the instruction cache; [Column 3, line 66 – column 4, line 10 and column 5, lines 26-29 show that the processor is a simultaneously redundant processor ("SRT", a type of simultaneous multithreaded processor that executes the same code in each thread) and thus there are at least two pipelines that execute the instructions in parallel as illustrated by figure 2 where the execution units 134, 138, and 142 are each replicated for such operation.]

c. a data value prediction module coupled to the second pipeline. [Column 6, line 59 – column 7, line 29 show that the SRT pipelines are separated by "slack" and thus the redundant threads comprise a leading (2^{nd}) and trailing (1^{st}) thread or pipeline. Column 7, lines 52-59 show that upon a [data] cache miss in the

leading thread or pipeline, the cache is updated so it will not miss in the trailing thread. This means that when a load instruction misses the cache in the leading thread the load value is retrieved from memory and updates the data cache. Thus a the trailing thread predicts that the trailing thread will use the same load value and updates the cache accordingly using an inherent module, which is appropriately called a data value prediction module. the given definition shows that to predict is to "make known in advance". Thus the leading thread by making the data available in the cache ahead of time or make the data know in advance by placing it in the cache for the trailing thread and thus the second pipeline makes this data known in advance or predicts this data.]

15. In regard to claim 15, Mukherjee discloses the computer of claim 13, further comprising: a bus coupled to the data cache and the memory, wherein the first processor includes the second pipeline. [Figure 1 shows that the memory (DRAM) is coupled to the processor via a bus and thus also to the data cache (directly or indirectly) for access on a cache miss. Figure 2 shows that both pipelines are in a single processor.]

16. In regard to claim 17, Mukherjee discloses the computer of claim 13, further comprising: a synchronization mechanism coupled to the second pipeline. [Column 7, lines 7-29 show that a slack counter is used to keep a delay between the two pipelines (threads) or synchronize them by a certain delay allowing the leading thread to run-ahead.]

17. In regard to claim 18, Mukherjee discloses the computer of claim 17, wherein the synchronization mechanism includes a run-ahead counter (as described above).

18. In regard to claim 19, Mukherjee discloses the computer of claim 13, further comprising: a storage buffer coupled to the second pipeline. [Figure 2, illustrates a storage buffer as element 130. Column 6, lines 15-23 show that this unit acts as an instruction queue or buffer.]

19. In regard to claim 20, Mukherjee discloses an article comprising a computer-readable medium having associated data (figure 2, element 110 holds instructions that cause the processor to perform the following limitations), wherein the medium causes a computer to perform the following:

a. executing a plurality of instructions including a LOAD instruction using a first pipeline sharing an instruction cache and a data cache with a second pipeline; [Column 6, lines 31-43 show that a plurality of instructions including load instructions are executed. Column 3, line 66 – column 4, line 10 and column 5, lines 26-29 show that the processor is a simultaneously redundant processor ("SRT", a type of simultaneous multithreaded processor that executes the same code in each thread) and thus there are at least two pipelines that execute the instructions in parallel as illustrated by figure 2 where the execution units 134, 138, and 142 are each replicated for such operation. Figure 2 also shows that these multiple pipelines use a single shared instruction cache and data cache.]

b. calculating a predicted load value for execution of the LOAD instruction if a cache miss in the data cache results when the second pipeline executes the

LOAD instruction before the first pipeline; [Column 6, line 59 – column 7, line 29 show that the SRT pipelines are separated by "slack" and thus the redundant threads comprise a leading (2^{nd}) and trailing (1^{st}) thread or pipeline. Column 7, lines 52-59 show that upon a [data] cache miss in the leading thread or pipeline, the cache is updated so it will not miss in the trailing thread. This means that when a load instruction misses the cache in the leading thread the load value is retrieved from memory and updates the data cache. These steps of retrieval and updating are being interpreted as being a calculation or operation. Thus a predicted load value is calculated for execution of the load instruction in the 1^{st} or trailing thread. the given definition shows that to predict is to "make known in advance". Thus the leading thread by making the data available in the cache ahead of time or make the data know in advance by placing it in the cache for the trailing thread and thus the second pipeline makes this data known in advance or predicts this data.]

c. and continuing execution of the plurality of instructions using the second pipeline. [As described in the sections above and with figure 3, the pipelines each execute the same program and after each instruction (including the cache-missed load instructions) the program continues as before so that the program may execute to completion.]

20. In regard to claim 22, Mukherjee discloses the article of claim 20, wherein the computer-readable medium further causes the computer to perform:

a. counting a number of instructions included in the plurality of instructions which the second pipeline has executed ahead of the first pipeline;

b. and restarting execution of the plurality of instructions by the second pipeline at a program counter value maintained by the first pipeline if the number of instructions is greater than or equal to a preselected threshold value.

As shown in sections cited previously, an amount of slack or number of instructions is kept between the two pipelines via a counter. The value of this counter is added to the program counter (making a separate or second program counter value) and if beyond the minimum amount of slack, the trailing thread re-executes the same instructions executed by the leading thread.

21. In regard to claim 23, Mukherjee discloses the article of claim 20, wherein the computer-readable medium further causes the computer to perform: beginning execution of the plurality of instructions by the first and second pipelines at a same program counter value. [As shown above, the pipelines are offset by a slack counter but both use the same base program counter value.]

22. In regard to claim 24, Mukherjee discloses a method of processing data extremely similar to the operation described in claim 20 and thus the same arguments exist for both claims.

23. In regard to claims 26 and 27, Mukherjee discloses a method of processing data extremely similar to the operation described in claims 22 and 23, respectively, and thus the same arguments exist for both claims.

Claim Rejections - 35 USC § 103

24. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

25. Claims 2-3, 5, and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Mukherjee.

26. In regard to claim 2,

a. Mukherjee discloses the data processing apparatus of claim 1.

b. Mukherjee does not disclose a first instruction fetch module coupled to the first pipeline and a second instruction fetch module coupled to the second pipeline.

c. While a plurality of fetch modules is not explicitly taught, a single instruction fetch module is taught in figure 2, element 102 of Mukherjee is taught. The inclusion of a plurality of fetch modules to perform the same function as a single fetch unit provides no new or unexpected result over the prior art. Therefore, one of ordinary skill in the art would have found it obvious to duplicate the fetch module, creating a fetch module for each pipeline (see MPEP 2144.04 (VI): In re Harza, 274 F.2d 669, 671, 124 USPQ 378, 380 (CCPA 1960)).

27. In regard to claim 3, Mukherjee discloses the data processing apparatus of claim 2, further comprising: a branch predictor (figure 2, element 103) coupled to the first and second instruction fetch modules.

28. In regard to claim 5,

a. Mukerhjee discloses the data processing apparatus of claim 1,

b. Mukherjee does not disclose wherein the first pipeline is included in a first processor, and wherein the second pipeline is included in a second processor.

c. While a separate processor for the second pipeline is not explicitly taught, multiple threads or pipelines of execution are taught by Mukherjee. The shifting of the location of the second pipeline into a second processor rather than within the same processor to perform the same exact function provides no new or unexpected result over the prior art. Therefore, one of ordinary skill in the art would have found it obvious to place the second pipeline in a separate processor (see MPEP 2144.04 (VI): In re Harza, 274 F.2d 669, 671, 124 USPQ 378, 380 (CCPA 1960)).

29. In regard to claim 14,

a. Mukerhjee discloses the computer of claim 13,

b. Mukherjee does not disclose wherein the first pipeline is included in a first processor, and wherein the second pipeline is included in a second processor.

c. While a separate processor for the second pipeline is not explicitly taught, multiple threads or pipelines of execution are taught by Mukherjee. The shifting of the location of the second pipeline into a second processor rather than within the same processor to perform the same exact function provides no new or unexpected result over the prior art. Therefore, one of ordinary skill in the art would have found it obvious to place the second pipeline in a separate processor

(see MPEP 2144.04 (VI): In re Harza, 274 F.2d 669, 671, 124 USPQ 378, 380 (CCPA 1960)).

30. Claims 8, 16, 21, and 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Mukherjee in view of Kaxiras (Improving CC-NUMA Performance...).

31. In regard to claim 8,

a. Mukherjee disclosed the data processing apparatus of claim 1.

b. Mukherjee has not disclosed a value prediction table coupled to the value prediction module.

c. Kaxiras has disclosed a value predictor for a cache miss occurrence with a load instruction (Page 1, column 2 and section 2, for example). Figure 1 illustrates that this prediction uses a table for looking up prediction values on a load miss.

d. The abstract of Kaxiras shows that this prediction method uses few hardware resources and gives the performance of address-based prediction, thus avoiding having to fetch the appropriate data value from memory in the leading thread. This performance gain coupled with the need for little additional hardware would have motivated one of ordinary skill in the art to modify the design of Mukherjee to use the value prediction table method given by Kaxiras. It would have been obvious to one of ordinary skill in the art at the time of invention to modify the design of Mukherjee to use the value prediction table method of value prediction disclosed by Kaxiras so that performance gains are realized without adding much additional hardware.

32. In regard to claim 16,

a. Mukherjee disclosed the computer of claim 13.

b. Mukherjee has not disclosed a value prediction table coupled to the value prediction module.

c. Kaxiras has disclosed a value predictor for a cache miss occurrence with a load instruction (Page 1, column 2 and section 2, for example). Figure 1 illustrates that this prediction uses a table for looking up prediction values on a load miss.

d. The abstract of Kaxiras shows that this prediction method uses few hardware resources and gives the performance of address-based prediction, thus avoiding having to fetch the appropriate data value from memory in the leading thread. This performance gain coupled with the need for little additional hardware would have motivated one of ordinary skill in the art to modify the design of Mukherjee to use the value prediction table method given by Kaxiras.

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the design of Mukherjee to use the value prediction table method of value prediction disclosed by Kaxiras so that performance gains are realized without adding much additional hardware.

33. In regard to claim 21,

a. Mukherjee disclosed the article of claim 20 where the second pipeline restarts execution of that which was executed in the first pipeline with any

optimizations made for faster processing based on a program counter value plus the slack counter value all as disclosed above.

b. Mukherjee has not disclosed

i. counting a number of mispredictions occurring when the predicted load value is incorrect;

ii. and restarting execution of the plurality of instructions by the second pipeline at a program counter value maintained by the first pipeline if the number of mispredictions is greater than or equal to a preselected threshold value.

c. Kaxiras has disclosed in section 4.1 under the heading "a working example" the use of a counter to count mispredictions up to a threshold value, where correct predictions are then made.

d. The abstract of Kaxiras shows that this prediction method uses few hardware resources and gives the performance of address-based prediction, thus avoiding having to fetch the appropriate data value from memory in the leading thread. This performance gain coupled with the need for little additional hardware would have motivated one of ordinary skill in the art to modify the design of Mukherjee to use the misprediction counter method given by Kaxiras. With this prediction scheme in place, the trailing pipeline of Mukherjee re-executes the instructions run by the leading thread but optimized with the value predictions made using the misprediction counter.

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the design of Mukherjee to use the misprediction counter method of value prediction disclosed by Kaxiras so that performance gains are realized without adding much additional hardware.

34. In regard to claim 25, the claim limitations are extremely similar to those of claim 21 and thus the same arguments apply.

Response to Arguments

35. Applicant's arguments filed 12/22/04 have been fully considered but they are not persuasive.

36. Applicant has argued with respect to claims 1 and 13 that Mukherjee does not teach or suggest a data value prediction module coupled to the second pipeline. Applicant states that instead Mukherjee allows the leading thread to resolve the cache miss so that the corresponding instructions in the trailing thread will not experience the cache miss and that there is thus no need in Mukherjee to have a data value prediction module. The Examiner directs Applicant to the included dictionary definition of the term "predict". This definition shows that to predict is to "make known in advance". Thus the leading thread by making the data available in the cache ahead of time or make the data know in advance by placing it in the cache for the trailing thread and thus the second pipeline, which makes this data known in advance or predicts this data, includes a data value prediction module.

37. Applicant has argued with respect to claims 20 and 24 that Mukherjee does not teach or suggest "calculating a predicted load value for execution of the LOAD

instruction if a cache miss in the data cache results when the second pipeline executes the LOAD instruction before the first pipeline". Applicant states that Mukherjee does not need to predict a load value because he actually resolves the cache miss with the leading thread as given in the preceding paragraph. Similar to the preceding paragraph, the given definition shows that to predict is to "make known in advance". Thus the leading thread by making the data available in the cache ahead of time or make the data know in advance by placing it in the cache for the trailing thread and thus the second pipeline makes this data known in advance or predicts this data.

38. Applicant has argued that the 35 USC 103 combination are improper due to lack of motivation because Mukherjee does not need to predict a load value as argued previously. Again, the given definition shows that to predict is to "make known in advance". Thus the leading thread by making the data available in the cache ahead of time or make the data know in advance by placing it in the cache for the trailing thread and thus the second pipeline makes this data known in advance or predicts this data.

Conclusion

39. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any

extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

40. The following is text cited from 37 CFR 1.111(c): In amending in reply to a rejection of claims in an application or patent under reexamination, the applicant or patent owner must clearly point out the patentable novelty which he or she thinks the claims present in view of the state of the art disclosed by the references cited or the objections made. The applicant or patent owner must also show how the amendments avoid such references or objections.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Shane F Gerstl whose telephone number is (571) 272-4166. The examiner can normally be reached on M-F 6:45-4:15 (First Friday Off).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Shane F Gerstl
Examiner
Art Unit 2183

SFG
January 6, 2005

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100